# Power Efficient and Sparse Spanner for Wireless Ad Hoc Networks

Xiang-Yang Li\*   Peng-Jun Wan\*   Yu Wang\*

*Abstract*—**Due to the limited resources available in the wireless ad hoc networking nodes, the scalability is crucial for network operations. One effective approach is to maintain only a sparse spanner of a linear number of links while still preseving the power-efficient route for any pair of nodes. For any spanner $G$, its power stretch factor is defined as the maximum ratio of the minimum power needed to support any link in this spanner to the least necessary. In this paper, we first consider several well-known proximity graphs including relative neighborhood graph, Gabriel graph and Yao graph. These graphs are sparse and can be constructed locally in an efficient way. We show that the power stretch factor of Gabriel graph is always one, and the power stretch factor of Yao graph is bounded by a constant while the power stretch factor of relative neighborhood graph could be as large as the network size minus one. Notice that all of these graphs do not have constant degrees. We further propose another sparse spanner that has both constant degree and constant power stretch factor. An efficient local algorithm is presented for the construction of this spanner.**

*Keywords*—**Wireless ad hoc networks, topology control, power consumption, network optimization.**

## I. INTRODUCTION

We consider a wireless ad hoc network consisting of a set of nodes distributed in a two-dimensional plane. Each node has an omnidirectional antenna. In the most common power-attenuation model, the power needed to support a link $uv$ is $\|uv\|^\beta$, where $\|uv\|$ is the distance between $u$ and $v$, $\beta$ is a constant between 2 and 4 dependent on the wireless transmission environment. By proper scaling, all nodes have maximum transmission range equal to one unit. These nodes define a *unit disk graph* in which there is an edge between two nodes if and only if their distance is at most one. The number of edges in the unit disk graph could be as large as the square order of the number of network nodes. The routing over this unit disk graph is unscalable.

Recently, Rodoplu and Meng [1] described a distributed protocol to construct a spanner, a connected subgraph of the unit disk graph, which is guaranteed to contain the minimum energy consumption path connecting any pair of nodes. However, their protocol is not time and space efficient. In the worst case, the time complexity of each node is in the cubic order of the number of its neighbors. Recently, improvement on their result was made in [2] to construct a sparser spanner more efficiently. The constructed spanner has a linear number of edges and also preserve the path of the minimum power for any pair of nodes.

A further trade-off can be made between the sparseness and the power efficiency of the spanner. The power efficiency of any spanner is measured by its power stretch factor, which is defined as the maximum ratio of the minimum power needed to support any link in this spanner to the least necessary. Recently, Wattenhofer *et al.* [3] try to address this trade-off. Unfortunately, their algorithm is problematic and their result is erroneous which will be discussed in detail later.

In this paper, we first consider several well-known proximity graphs including relative neighborhood graph, Gabriel graph and Yao graph. These graphs are sparse and can be constructed locally in an efficient way. We show that the power stretch factor of Gabriel graph is always one, and the power stretch factor of Yao graph is bounded by a constant while the power stretch factor of relative neighborhood graph could be as large as the network size minus one. Notice that all of these graphs do not have constant degrees. We further propose another sparse topology that has both constant degree and constant power stretch factor. An efficient local algorithm is presented for constructing this topology.

\* Department of Computer Science, Illinois Institute of Technology, Chicago, IL 60526

The rest of the paper is organized as follows. In Section II, we review the definitions of several well-known geometry structures. We formally define the power stretch factor and introduce several basic properties in Section III. In Section IV, we study several geometry structures that may be used for network topology control. We then give an algorithm that constructs a sparse and power-efficient spanner with bounded degree. We conclude our paper in section V by discussing some possible future works.

## II. GEOMETRY STRUCTURES

Let $V$ be a set of $n$ wireless nodes distributed in a two-dimensional plane. These nodes induce a *unit disk graph* $UDG(V)$ in which there is an edge between two nodes if and only if their distance is at most one. Various proximity subgraphs of the unit disk graph can be defined.

- The *constrained relative neighborhood graph*, denoted by $RNG(V)$, consists of all edges $uv$ such that $\|uv\| \leq 1$ and there is no point $w \in V$ such that $\|uw\| < \|uv\|$, and $\|wv\| < \|uv\|$.
- The *constrained Gabriel graph*, denoted by $GG(V)$, consists of all edges $uv$ such that $\|uv\| \leq 1$ and the open disk using $uv$ as diameter does not contain any node from $V$.
- The *constrained Yao graph* with an integer parameter $k \geq 6$, denoted by $\overrightarrow{YG}_k(V)$, is defined as follows. At each node $u$, any $k$ equal-separated rays originated at $u$ defined $k$ cones. In each cone, choose the closest node $v$ to $u$ with distance at most one, if there is any, and add a directed link $\overrightarrow{uv}$. Ties are broken arbitrarily. Let $YG_k(V)$ be the undirected graph by ignoring the direction of each link in $\overrightarrow{YG}_k(V)$. See the left figure of Figure 1 for an illustration.
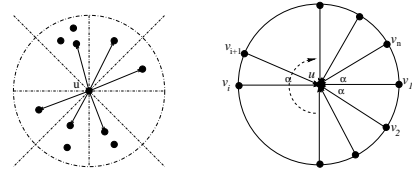


Fig. 1. Left: The narrow regions are defined by 8 equal cones and the closest node in each cone is a neighbor of $u$; Right: Node $u$ in the $RNG$ has degree $n$.

These graphs extend the conventional definitions of corresponding ones for the completed Euclidean graph. It is well-known that $RNG(V)$ is a subgraph of $GG(V)$ and $YG_k(V)$ [4], [5], [6]. In addition, all these graphs contain the Euclidean minimum spanning tree $EMST(V)$ as a subgraph. These graphs are sparse: $|RNG(V)| \leq 3n - 10, |GG(V)| \leq 3n - 8$, and $|\overrightarrow{YG}_k(V)| \leq kn$.

The sparseness implies that the average node degree is bounded by a constant. However the maximum degree could be as large as $n - 1$ as shown in the right figure of Figure 1. The instance consists of $n$ points lying on the unit circle centered at a node $u \in V$. Thus, each edge $uv_i$ belongs to the $RNG(V), GG(V)$ and $YG_k(V)$.

The configuration given by the right figure of Figure 1 shows that no geometry structure with constant degree bound contains the least energy consumption path for all pairs of nodes. Notice that if such structure exists, node $u$ has to maintain the connection $uv_i$ because $uv_i$ is the minimum energy consumption path for $u$ and $v_i$.

The length stretch factor (Some researchers call it *dilation ratio*, *spanning ratio*) of a graph $G$ is defined as the maximum ratio of the shortest path length connecting any pair of nodes in $G$ to their distance. Bose *et al.* [7] showed that the length stretch factor of $RNG(V)$ is at most $n-1$ and the length stretch factor of $GG(V)$ is at most $\frac{4\pi\sqrt{2n-4}}{3}$. Several papers showed that the Yao graph $YG_k(V)$ has length stretch factor at most $\frac{1}{1-2\sin\frac{\pi}{k}}$.

## III. POWER STRETCH FACTOR

Consider any unicast path $\pi(u,v)$ in $G$ (could be directed) from a node $u \in V$ to another node $v \in V$, say $\pi(u,v) = v_0 v_1 \cdots v_{h-1} v_h$, where $u = v_0$, $v = v_h$. The total *transmission power* $p(\pi)$ consumed by this path $\pi$ is $p(\pi) = \sum_{i=1}^{h} \|v_{i-1} v_i\|^\beta$. Let $p_G(u,v)$ be the least energy consumed by all paths connecting nodes $u$ and $v$ in $G$. The path connecting $u, v$ and consuming the energy $p_G(u,v)$ is called the *least-energy path* in $G$ for $u$ and $v$. When it is clear from the context, we will omit the subscript $G$ in $p_G(u,v)$.

Let $H$ be a subgraph of $G$. The *power stretch factor* of a graph $H$ with respect to $G$ is then defined as

$$\rho_H(G) = \max_{u,v \in V} \frac{p_H(u,v)}{p_G(u,v)}$$

If $G$ is a unit disk graph, we use $\rho_H(V)$ instead of $\rho_H(G)$. Let

$$\rho_H(n) = \sup_{|V|=n} \rho_H(V).$$

When the graph $H$ is clear from the context, it is dropped from notation. In this section, we present some basic properties of power stretch factor.

*Lemma 1:* For a constant $\delta$, $\rho_H(G) \leq \delta$ iff for any link $v_i v_j$ in graph $G$ but not in $H$, $p_H(v_i, v_j) \leq \delta \|v_i v_j\|^\beta$.
Proof. The necessary part is obvious. We only concentrate on the sufficient part. Consider any two nodes $u$ and $v$. Consider the least power consumption path $\pi_G(u,v) = v_0 v_1 \cdots v_{h-1} v_h$, where $u = v_0$, $v = v_h$. Then for each link $v_i v_{i+1}$, there is a path $\pi_H(v_i, v_{i+1})$ in $H$ with energy consumption at most $\delta \|v_i v_{i+1}\|^\beta$. Consider the path formed by concatenating all paths $\pi_H(v_i, v_{i+1})$, $i = 0, \cdots, h-1$. Its power consumption is at most $\sum_{i=1}^{h} p(\pi_H(v_i, v_{i+1})) \leq \sum_{i=1}^{h}(\delta \|v_i v_{i+1}\|^\beta) = \delta \cdot p_G(u,v)$. Then the lemma follows.  ▣

The above lemma implies that it is sufficient to analyze the power stretch factor of $H$ to each link in $G$ but not in $H$.

*Lemma 2:* For any $H \subseteq G$ with length stretch factor $\delta$, its power stretch factor is at most $\delta^\beta$ for any graph $G$.
Proof. From Lemma 1, it is sufficient to show $p_H(u,v) \leq \delta^\beta \|uv\|^\beta$ for any link $uv$ in $G$ but not in $H$. There is a path $\pi_H(u,v)$ in $H$ with length at most $\delta \|uv\|$. The lemma follows from $p(\pi_H(u,v)) = \sum_{e \in \pi_H(u,v)} \|e\|^\beta \leq \left(\sum_{e \in \pi_H(u,v)} \|e\|\right)^\beta \leq (\delta \|uv\|)^\beta$.  ▣

Therefore a geometry structure $H$ with a constant length stretch factor $\delta$ implies that its power stretch factor is no more than $\delta^\beta$. In particular, a graph with a constant length stretch factor must also have a constant power stretch factor. But the reverse is not necessarily true. Finally, the power stretch factor has the following monotonic property, i.e., $H_1 \subset H_2 \subset G$ then $\rho_{H_1}(G) \geq \rho_{H_2}(G)$.

## IV. RESULTS

In this section, we study the power stretch factor of several sparse geometry structures for unit graph although our results usually hold for general graphs. At the end, we give a method to construct a sparse network with bounded degree and bounded stretch factor.

### A. Relative Neighborhood Graph

Since the relative neighborhood graph has the length stretch factor as large as $n-1$, then the Lemma 2 implies that its power stretch factor is at most $(n-1)^2$. In this section, we show that it is actually $n-1$.

*Theorem 3:* $\rho_{RNG}(n) = n-1$.
Proof. First we prove that $\rho_{RNG}(n)$ is at most $n-1$. Consider the path between $u$ and $v$ in $EMST(V)$. This path contains at most $n-1$ edges and each edge has length at most $\|uv\|$. Thus, its total power consumption is at most $(n-1)\|uv\|^\beta$. Notice $EMST(V) \subset RNG(V)$ if $UDG(V)$ is connected. From Lemma 1, $\rho_{RNG}(n) \leq n-1$.

Then we show that $\rho_{RNG}(n) \geq n-1-\varepsilon$ for any small positive $\varepsilon$ by constructing an example illustrated in Figure 2.
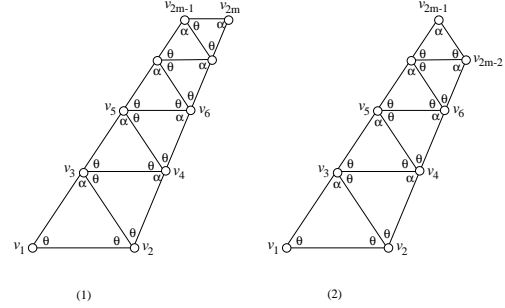


Fig. 2. The Euclidean minimum spanning tree has large stretch factor.

We consider two cases. We first consider even $n$, say $n = 2m$. The construction of the point set $V$ is shown in Figure 2 (1), which was used in [7]. Let $\alpha = \frac{\pi}{3} + 2\delta, \theta = \frac{\pi}{3} - \delta$, where $\delta$ is a sufficiently small positive number which will be fixed later. The $m$ points with odd subscripts $v_1, v_3, v_5, \cdots, v_{2m-1}$ are collinear, so are the $m$ points with even subscripts $v_2, v_4, v_6, \cdots, v_{2m}$. As proved in [7], $RNG(V)$ is a path $v_1, v_3, v_5, \cdots, v_{2m-1}, v_{2m}, \cdots, v_6, v_4, v_2$. As $\delta \longrightarrow 0$, the length of each edge in $RNG(V)$ tends to $\|v_1 v_2\|$ from below, which implies $\frac{p_{RNG}(u,v)}{p(u,v)} \longrightarrow n-1$. So we can find a sufficiently small $\delta > 0$ such that $\frac{p_{RNG}(u,v)}{p(u,v)} > n-1-\epsilon$, which implies $\rho_{RNG}(n) > n-1-\epsilon$.
When $n$ is odd, the construction is shown in Figure 2 (2) and the existence can be proved by a similar argument.  ▣

The above proof also shows that any graph contains the Euclidean minimum spanning tree has the power stretch factor at most $n-1$.

### B. Gabriel Graph

The Gabriel graph has length stretch factor in between $\frac{\sqrt{n}}{2}$ and $\frac{4\pi\sqrt{2n-4}}{3}$[7]. From Lemma 2, its power stretch factor is at most $\left(\frac{4\pi\sqrt{2n-4}}{3}\right)^2$. We show that $\rho_{GG}(n) = 1$ by giving a stronger result.

*Theorem 4:* The power stretch factor of any Gabriel graph is one.
Proof. Consider any link $uv$ in any least energy path in $UDG(V)$. Then $uv$ itself is the least energy path between $u$ and $v$. Therefore, the open disk using $u$ and $v$ as diameter is empty of wireless nodes. It implies that edge $uv$ remains in the Gabriel graph $GG(V)$.  ▣

### C. Yao Graph

The Yao graph $YG_k(V)$ has length stretch factor $\frac{1}{1-2\sin\frac{\pi}{k}}$. Lemma 2 implies its power stretch factor is no more than $(\frac{1}{1-2\sin\frac{\pi}{k}})^\beta$. We prove a stronger result.

*Theorem 5:* The power stretch factor of the Yao graph $YG_k(V)$ is at most $\frac{1}{1-(2\sin\frac{\pi}{k})^\beta}$.

Proof. From Lemma 1, it is sufficient to show that for any nodes $u$ and $v$ with $\|uv\| \leq 1$, there is a path connecting $u$ and $v$ in $YG_k(V)$ with energy consumption at most $\frac{1}{1-(2\sin\frac{\pi}{k})^\beta}\|uv\|^\beta$. Let $\delta = \frac{1}{1-(2\sin\frac{\pi}{k})^\beta}$. We construct a path $u \rightsquigarrow v$ connecting $u$ and $v$ in $YG_k(V)$. If link $vu \in YG_k(V)$, then set the path $u \rightsquigarrow v$ as the link $uv$. Otherwise, there must exist another node $w$ in the same cone as $v$, which is a neighbor of $u$ in $YG_k(V)$. Then $u \rightsquigarrow v$ is set as the concatenation of the link $uw$ and path $w \rightsquigarrow v$. Notice that the angle $\theta$ of each cone section is $\frac{2\pi}{k}$. When $k > 6$, then $\theta < \frac{\pi}{3}$. It is easy to show that $\|wv\| < \|uv\|$. Consequently each node appears at most once in the path $u \rightsquigarrow v$. We prove by induction that the path $u \rightsquigarrow v$ has cost $p(u \rightsquigarrow v)$ at most $\delta\|uv\|^\beta$ on the number of its edges.

If $u \rightsquigarrow v = uv$, $p(u \rightsquigarrow v) = \|uv\|^\beta < \delta\|uv\|^\beta$. Assume that the claim is true for any path with $l$ edges. Then consider a path $u \rightsquigarrow v$ with $l+1$ edges, which is the concatenation of edge $uw$ and the path $w \rightsquigarrow v$ with $l$ edges. We consider two cases.

Case 1: the angle $\angle uwv$ is not acute. We have $\|uw\|^2 + \|wv\|^2 \leq \|uv\|^2$. Notice that $\frac{\|uw\|}{\|uv\|} \leq 1$ and $\frac{\|wv\|}{\|uv\|} \leq 1$. It implies that

$$\left(\frac{\|uw\|}{\|uv\|}\right)^\beta + \left(\frac{\|wv\|}{\|uv\|}\right)^\beta \leq \left(\frac{\|uw\|}{\|uv\|}\right)^2 + \left(\frac{\|wv\|}{\|uv\|}\right)^2 \leq 1$$

Therefore, $\|uw\|^\beta + \|wv\|^\beta \leq \|uv\|^\beta$ for any $\beta \geq 2$. Notice that $\|wv\| < \|uv\| \leq 1$, which implies that we can apply induction on the path $w \rightsquigarrow v$ also. See the left figure of Figure 3. Thus, $p(w \rightsquigarrow v) \leq \delta\|wv\|^\beta$ by induction. Then
$p(u \rightsquigarrow v) = \|uw\|^\beta + p(w \rightsquigarrow v) \leq \|uw\|^\beta + \delta\|wv\|^\beta \leq \delta\|uv\|^\beta$.
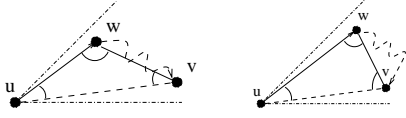


Fig. 3. Left: The angle $\angle uwv$ is not acute; Right: The angle $\angle uwv$ is acute.

Case 2: the angle $\angle uwv$ is acute. See the right figure of Figure 3. Notice $\|uw\| \leq \|uv\|$ and $\angle wuv < \theta$ because of the definition of the neighbors in the narrow region graph. The maximum length of $vw$ is achieved when $\|uw\| = \|uv\|$ because the angle $\angle uwv$ is acute. Therefore $\|wv\| \leq 2\sin\frac{\theta}{2}\|uv\| = 2\sin\frac{\pi}{k}\|uv\|$. By induction, we have $p(u \rightsquigarrow v) \leq \|uw\|^\beta + \delta\|wv\|^\beta$, which is at most

$$\|uv\|^\beta + \frac{1}{1-(2\sin\frac{\pi}{k})^\beta}(2\sin\frac{\pi}{k})^\beta\|uv\|^\beta = \frac{1}{1-(2\sin\frac{\pi}{k})^\beta}\|uv\|^\beta.$$

This finishes the proof. ▱

We end this section by commenting a result by Wattenhofer *et al.* [3]. Their two-phased approach consists of a variation of the Yao graph followed by a variation of the Gabriel graph. They try to prove that the constructed spanner has constant power stretch factor and the node degree is bounded by a constant. Unfortunately, there are some bugs in their proof of the constant power stretch factor and their result is erroneous. Their algorithm can be interpreted as follows.
• The first phase constructs a variation of the Yao graph. For two nodes $u$ and $v$ with $\|uv\| \leq 1$, define the cone $cone(u, v, \theta)$ as the wedge with angle $\theta$ and $uv$ as the bisector. Each node $u$ maintains a neighbors set $N(u)$ which is initialized to empty. A node $v$ is added to $N(u)$ if $cone(u, v, \theta)$ is not fully contained in $\cup_{w \in N(u)} cone(u, w, \theta)$. The construction ends either all neighbors are scanned or the $\cup_{w \in N(u)} cone(u, w, \theta)$ cover the entire two-dimensional space, whichever comes first. Then node $v \in N(u)$ also puts $u$ in its neighbor set.
• The second phase constructs a variation of the Gabriel graph over the graph constructed from the first phase. For any node $u$, if there are two

nodes $v$ and $w$ with $v, w \in N(u)$ and $w \in N(v)$ such that

$$p(u, v) + p(v, w) \leq q \cdot p(u, w)$$

for a constant $q$, where $q$ is a parameter chosen by the algorithm, then remove $w$ from $N(u)$ (and by symmetry, $w$ also removes $u$ from $N(w)$). If there is more than one node $v$ satisfies the power inequality for node $w$, then choose the node with the minimum distance to $u$. Their algorithm traverses the neighbor nodes in the order of the distance to $u$. This makes sure that node $v$ will stay.

They proved that after the first phase, the power stretch factor of the constructed graph is at most $1 + 2\sin\frac{\theta}{2}$. Their proof contains several bugs and therefore fail to obtain the claimed power stretch factor. The bound $1 + 2\sin\frac{\theta}{2}$ also appears to be erroneous. Using our approach, we can fix this bug and obtain a power stretch factor $\frac{1}{1-(2\sin\frac{\theta}{2})^\beta}$.

We also observe that the second phase of the algorithm is not well-defined and contain serious bugs that can not be fixed. The algorithm depends on the relative order of removing edges.
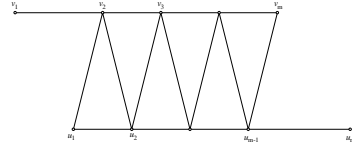


Fig. 4. The algorithm may be ill-functional.

Consider the instance shown in Figure 4. It consists of $n = 2m$ nodes $V = \{u_i : 1 \leq i \leq m\} \cup \{v_i : 1 \leq i \leq m\}$. Nodes $\{u_i : 1 \leq i \leq m\}$ and $\{v_i : 1 \leq i \leq m\}$ are distributed in two parallel lines with

$$\|u_iv_i\| = 1, 2 \leq i \leq m-1; \text{ and } \|u_iv_{i+1}\| = 1, 1 \leq i \leq m-1;$$
$$\|u_{i+1}v_i\| = 1, 1 \leq i \leq m-1; \text{ and } \|v_1v_2\| = \|u_{m-1}u_m\| = a+\epsilon$$
$$\|u_iu_{i+1}\| = \|v_{i+1}v_{i+2}\| = a, 1 \leq i \leq m-1;$$

where $a$ is a parameter satisfying $a^\beta < q-1$, and $\epsilon$ is a positive parameter such that $a + \epsilon < 1$. After the first phase, the $YG(V)$ consists of solid lines shown in Figure 4.

During the second phase, all the horizontal edges will survive. The other edges may be removed in the following order. Node $u_1$ will remove edge $u_1v_2$ due to the presence of node $u_2$. Node $v_2$ then remove the edge $v_2u_2$ due to the presence of node $v_3$. Node $u_2$ then remove edge $u_2v_3$ due to the presence of node $u_3$. The process continues until all slant edges except the edge $u_{m-1}v_m$. The resulting graph consists of all the horizontal links plus the edge $u_{m-1}v_m$. In this graph, the unique path from $u_1$ to $v_2$ has total power consumption $2(m-2)a^\beta+1$. Then the power stretch factor is at least $2(m-2)a^\beta+1$, which increases linearly with the number of nodes.

The key mistake in the algorithm is that each node may eliminate both incoming links and outgoing links.

### D. Bounded Degree Graph

Notice that even the directed graph $\overrightarrow{YG}_k(V)$ has bounded stretch ratio and bounded out-degree $k$ for each node, some nodes may have very large in-degree. The nodes configuration given by the right figure of Figure 1 will result a very large in-degree for node $u$. Bounded out-degree gives us advantages when apply several routing algorithms. However, unbounded in-degree at node $u$ will often cause large overhead at $u$. Therefore it is often imperative to construct a sparse network topology such that both the in-degree and the out-degree are bounded by a constant while it is still power-efficient.

Arya *et al.* [8] had given an ingenious technique to generate a bounded degree graph with constant length stretch factor. We apply the same technique to construct a sparse network topology with bounded degree and bounded power stretch factor. The technique is to replace the directed star consisting of all links towards a node $u$ by a directed tree $T(u)$ of a bounded degree with $u$ as the sink. Tree $T(u)$ is constructed recursively.

First, we compute the graph $\overrightarrow{YG}_k(V)$. Each node $u$ will have a set of in-coming nodes $I(u) = \{v \mid \overrightarrow{vu} \in \overrightarrow{YG}_k(V)\}$. Choose the same $k$ equal-sized cones centered at $u$: $C_1(u), C_2(u), \cdots C_k(u)$. In each of the $k$ cones, node $u$ finds the nearest node $y_i \in I(u)$, $1 \le i \le k$, if there is any. Link $\overrightarrow{y_i u}$ is added to $T(u)$ and $y_i$ is removed from $I(u)$. For any newly added $v$ in the order of their appearance in $T(u)$, let $w$ be its parent in $T(u)$ and choose the nearest node of $I(u) \cap C_i(w) - \{v\}$ in the cone $C_i(v)$, $1 \le i \le k$, centered at $v$. Then create a directed links from the found nodes to $v$ and remove the found nodes from $I(u)$. The process is terminated when $I(u)$ becomes empty. Here $u$ can construct the tree $T(u)$ and then broadcast the structure of $T(u)$ to all nodes in $T(u)$. Figure 5 illustrates a directed star centered at $u$ and the directed tree $T(u)$ constructed to replace the star.
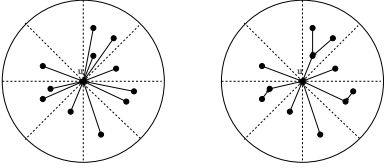


Fig. 5. Left: The directed star formed by all links towards to $u$; Right: The directed tree $T(u)$ sinked at $u$.

The union of all trees $T(u)$ is called the sink structure $\overrightarrow{YG}_k^*(V)$. We prove that its power stretch factor is at most $(\frac{1}{1-(2\sin\frac{\pi}{k})^\beta})^2$ and its degree is bounded by $(k+1)^2 - 1$.

*Theorem 6:* The power stretch factor of the graph $\overrightarrow{YG}_k^*(V)$ is at most $(\frac{1}{1-(2\sin\frac{\pi}{k})^\beta})^2$. The maximum degree of the graph $\overrightarrow{YG}_k^*(V)$ is at most $(k+1)^2 - 1$.

Proof. Using the same argument as Lemma 5, we can prove that for each node $v \in I(u)$, there is a direct path $\pi_{T(u)}(v, u)$ in $T(u)$ such that the power consumption of $\pi_{T(u)}(v, u)$ is no more than $\frac{1}{1-(2\sin\frac{\pi}{k})^\beta}\|vu\|^\beta$. It implies that the power stretch factor of the graph $\overrightarrow{YG}_k^*(V)$ is at most $(\frac{1}{1-(2\sin\frac{\pi}{k})^\beta})^2$.

Notice that the $snk$ geometry structure does not change the out-degree of a node. One directed edge $\overrightarrow{vu}$ implies that there is also one directed edge $\overrightarrow{vw}$ in the sink $T(u)$ for some $w \in I(u)$. Moreover, because each node $v$ at most participates in at most $k + 1$ sink trees (it will at most participate $k$ sink trees for some other nodes and itself will also have one sink tree $T(v)$) and $v$ participates in one sink tree will introduce at most $k$ in-degree, the total in-degree is therefore at most $k(k + 1)$. Consequently, the total degree is at most $k(k+1) + k = (k+1)^2 - 1$. □

Notice that the $snk$ structure and the Yao graph structure do not have to have the same number of cones. For setting up a power-efficient wireless networking, each node $u$ finds all its neighbors in $YG_k(V)$, which can be done in linear time proportional to the number of nodes within its transmission range.

A post-processing can be used to further reduce the number of edges. We apply the Gabriel graph structure to the constructed Yao graph $\overrightarrow{YG}_k(V)$: an edge $\overrightarrow{uv}$ in $\overrightarrow{YG}_k(V)$ survives if and only if there is no edge $\overrightarrow{uw}$ and $\overrightarrow{wv}$ in $\overrightarrow{YG}_k(V)$ and $\max(\|uw\|, \|uw\|) < \|uv\|$. The

constructed spanner still have the same power stretch factor and also with bounded out-degree $k$. Figure 6 illustrates different topologies defined in this paper for the unit disk graph illustrated by the first figure of Figure 6.
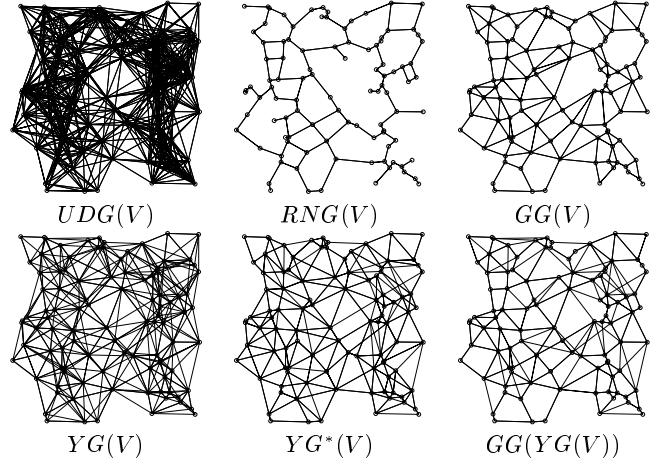


Fig. 6. Different topologies generated from the same *unit disk graph* $UDG(V)$.

## V. SUMMARY AND FUTURE WORK

In this paper, we consider how to maintain a simple network topology that is power-efficient. We summarize our results about the power stretch factors and some previous results about the length stretch factors by Table I.

|  | Length | Power | Degree |
|---|---|---|---|
| RNG | $n-1$ | $n-1$ | $n-1$ |
| GG | $\frac{4\pi\sqrt{2n-4}}{3}$ | $1$ | $n-1$ |
| YG | $\frac{1}{1-2\sin\frac{\pi}{k}}$ | $\frac{1}{1-(2\sin\frac{\pi}{k})^\beta}$ | $k, n-1$ |
| YG* | $\left(\frac{1}{1-2\sin\frac{\pi}{k}}\right)^2$ | $\left(\frac{1}{1-(2\sin\frac{\pi}{k})^\beta}\right)^2$ | $(k+1)^2-1$ |

TABLE I
THE DILATION RATIO OF DIFFERENT GRAPHS.

A node in Yao graph $\overrightarrow{YG}_k(V)$ has out-degree at most $k$, while its in-degree could be as large as $n-1$. Graph $YG^*(V)$ has a bounded degree and a constant power stretch factor. It is easy to give examples that the relative neighborhood graph, the Gabriel graph, the Yao graph and the sink struture could consume arbitrarily larger total energy by all nodes than the minimum total energy necessary to maintain the connectivity of the network. We leave it as a future work to design an algorithm to construct a connected topology with total energy consumption being a constant factor of the minimum necessary while achieving a bounded degree and a constant power stretch factor.

## REFERENCES

[1] Volkan Rodoplu and Teresa H. Meng, "Minimum energy mobile wireless networks," in *Proceedings of the 1998 IEEE International Conference on Communications, ICC'98*, 1998, vol. 3.
[2] Xiang-Yang Li and Peng-Jun Wan, "Constructing minimum energy mobile wireless networks," Submitted to for publication, 2001.
[3] Roger Wattenhofer, Li Li, Paramvir Bahl, and Yi-Min Wang, "Distributed topology control for wireless multihop ad-hoc networks," in *IEEE INFOCOM'01*, 2001.
[4] H. N. Gabow, J. L. Bently, and R. E. Tarjan, "Scaling and related techniques for geometry problems," in *ACM Symposium on Theory of Computing*, 1984, pp. 135–143.
[5] J. Katajainen, "The region approach for computing relative neighborhood graphs in the lp metric," *Computing*, vol. 40, pp. 147–161, 1988.
[6] A. C.-C. Yao, "On constructing minimum spanning trees in k-dimensional spaces and related problems," *SIAM J. Computing*, vol. 11, pp. 721–736, 1982.
[7] P. Bose, L. Devroye, W. Evans, and D. Kirkpatrick, "On the spanning ratio of gabriel graphs and beta-skeletons," Submitted to SIAM Journal on Discrete Mathematics, 2001.
[8] S. Arya, G. Das, D. Mount, J. Salowe, and M. Smid, "Euclidean spanners: short, thin, and lanky," in *Proc. 27th ACM STOC*, 1995, pp. 489–498.